

Merge Frame Design for Video Stream Switching using Piecewise Constant Functions

Wei Dai *Student Member, IEEE*, Gene Cheung *Senior Member, IEEE*, Ngai-Man Cheung *Senior Member, IEEE*, Antonio Ortega *Fellow, IEEE*, Oscar C. Au *Fellow, IEEE*

Abstract—The ability to efficiently switch from one pre-encoded video stream to another (e.g., for bitrate adaptation or view switching) is important for many interactive streaming applications. Recently, stream-switching mechanisms based on distributed source coding (DSC) have been proposed. In order to reduce the overall transmission rate, these approaches provide a “merge” mechanism, where information is sent to the decoder such that the exact same frame can be reconstructed given that any one of a known set of side information (SI) frames is available at the decoder (e.g., each SI frame may correspond to a different stream from which we are switching). However, the use of bit-plane coding and channel coding in many DSC approaches leads to complex coding and decoding. In this paper, we propose an alternative approach for merging multiple SI frames, using a piecewise constant (PWC) function as the merge operator. In our approach, for each block to be reconstructed, a series of parameters of these PWC merge functions are transmitted in order to guarantee identical reconstruction given the known side information blocks. We consider two different scenarios. In the first case, a target frame is first given, and then merge parameters are chosen so that this frame can be reconstructed exactly at the decoder. In contrast, in the second scenario, the reconstructed frame and merge parameters are jointly optimized to meet a rate-distortion criteria. Experiments show that for both scenarios, our proposed merge techniques can outperform both a recent approach based on DSC and the SP-frame approach in H.264, in terms of compression efficiency and decoder complexity.

I. INTRODUCTION

In conventional *non-interactive* video streaming, a client plays back successive frames in a pre-encoded stream in a fixed order. In contrast, in *interactive* video streaming [1], a client can switch freely in real-time among a number of pre-encoded streams. Examples include switching among multiple streams representing the same video encoded at different bit-rates for real-time bandwidth adaptation [2], or switching among views in a multi-view video [3]. See [1] for more examples of interactive streaming. A major challenge in interactive video streaming is to achieve efficient real-time switching among pre-encoded video streams. A simple approach would be to insert an intra-coded I-frame at each potential switching point [4]. But the relatively high rate required for I-frames often makes it impractical to insert them frequently in the streams, thus reducing the interactivity of playback.

W. Dai is with Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong. Email: weidai@connect.ust.hk

G. Cheung is with National Institute of Informatics, 2-1-2, Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430, Japan. Email: cheung@nii.ac.jp

N.-M. Cheung is with Singapore University of Technology and Design, 8 Somapah Road, Singapore 487372. Email: ngaiman_cheung@sutd.edu.sg

A. Ortega is with University of Southern California, 3740 McClintock Ave., Los Angeles, CA 90089-2564. Email: ortega@sipi.usc.edu

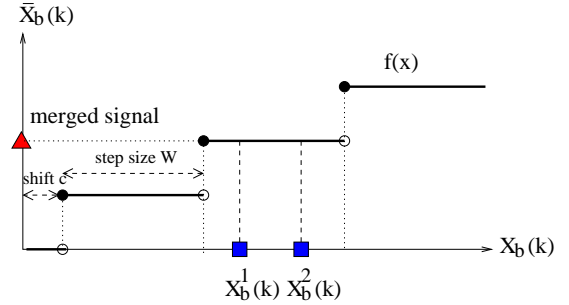


Fig. 1. Given the k -th coefficient $X_b(k)$ in block b from either SI frame 1 or 2, a piecewise constant function $f(x)$ maps either one ($X_b^1(k)$ or $X_b^2(k)$) to the same $\bar{X}_b(k)$ if they fall on the same constant interval.

Towards a more efficient stream-switching mechanism, *distributed source coding* (DSC) has been proposed. DSC can in principle achieve compression efficiency that is a function of the worst-case correlation between the target frame and the *side information* (SI) frames (from which the client may be switching) [5–7]. As an example, illustrated by Fig. 1, in the block-based DCT approach of [7], a desired k -th quantized frequency coefficient value $\bar{X}_b(k)$ in block b of the target frame is reconstructed using either $X_b^1(k)$ or $X_b^2(k)$, the corresponding coefficients in SI frames 1 and 2, respectively. A *D-frame* is transmitted so that it is possible to reconstruct the exact same target frame given any one of the two SI frames [7]. Thus we say that the D-frame supports a *merge* operation. In particular, the least significant bits (LSBs) of $X_b^1(k)$ and $X_b^2(k)$ are treated as “noisy” versions of the LSBs of $\bar{X}_b(k)$. The most significant bits (MSBs) of $\bar{X}_b(k)$ are obtained from the MSBs of $X_b^1(k)$ or $X_b^2(k)$, which are identical, while the D-frame contains channel codes that can produce the actual LSBs of $\bar{X}_b(k)$ taking $X_b^1(k)$ or $X_b^2(k)$ as inputs. The channel codes associated to these target frame coefficients compose the D-frames, which potentially require significantly fewer bits than an I-frame representation of the target frame [7].

There remain significant hurdles towards practical implementation of D-frames, however. First, the use of bit-plane encoding and channel codes in proposed techniques [7] means that the computation complexity at the decoder is high. Second, because the average statistics of a transform coefficient bit-plane for the entire image are used, non-stationary noise statistics can lead to high rate channel codes, resulting in coding inefficiency.

In this paper, we propose to use a *piecewise constant* (PWC)

function¹ as the signal merging operator. This approach operates directly on quantized frequency coefficients (instead of using a bit-plane representation) and does not require channel codes. As will be discussed in more detail in Section VI-C, our signal merging approach can be interpreted as a generalization of *coset coding* [9], where we explicitly optimize the merged target values for improved rate-distortion (RD) performance. The basic idea of our approach is summarized in Fig. 1, which depicts a *floor* function characterized by two parameters: a step size W and a shift c . In our approach, the encoder selects W and c to guarantee that $X_b^1(k)$ and $X_b^2(k)$ are in the same interval and thus map to the same reconstruction value. A W will be chosen for each frequency k , based on the statistics of the various $X_b(k)$ across all blocks b . Then, given W it will be possible to adjust c so that the reconstructed value matches a desired target, $\tilde{X}_b(k)$. A value of c will be chosen for each k and b , so that the bitrate required by our proposed technique is dominated by the cost of transmitting c . In this paper, we will formulate the problem of selecting c and W , and develop techniques for RD optimization of this selection.

We consider two scenarios. In the first one, *fixed target merging*, we will assume that $\tilde{X}_b(k)$ has been given, *e.g.*, by first generating an intra-coded version of the target frame, and using the corresponding quantized coefficient values as targets. We will show how to choose W to guarantee that $\tilde{X}_b(k)$ can be reconstructed. We will also show that given W , c is fixed. This type of merging is useful when there are cycles in the interactive playback, *i.e.*, frame A is an SI frame for frame B and B is an SI frame for A . This will be the case in *static view switching* for multiview video streaming, to be discussed in Section III.

In the second scenario, *optimized target merging*, we select W , c and $\tilde{X}_b(k)$ based on an RD criteria, where distortion is computed with respect to a desired target $X_b^0(k)$. In this scenario, we can use smaller values for W , and no longer need to select a fixed c for a given W and $\tilde{X}_b(k)$. This allows us to optimize c so as to significantly reduce the rate needed to encode the merging information. This approach can be used when there are no cycles in the interactive playback, *e.g.*, in *dynamic view switching* scenarios (also discussed in Section III). Experimental results show significant compression gains over D-frames [7] and SP-frames in H.264 [10] at reduced decoder computation complexity.

The paper is organized as follows. We first summarize related work in Section II. We then provide an overview of our coding system in Section III. We discuss the use of PWC functions for signal merging in Section IV. We present our PWC function parameter selection methods for fixed target merging and optimized target merging in Section V and VI, respectively. Finally, we present experimental results and conclusions in Section VII and VIII, respectively.

II. RELATED WORK

The H.264 video coding standard [11] introduced the concept of *SP-frames* [10] for stream-switching. In a nutshell, first the difference between one SI frame and the target picture is

lossily coded as the primary SP-frame. Then, the difference between each additional SI frame and the reconstructed primary SP-frame is *losslessly* coded as a secondary SP-frame; lossless coding ensures identical reconstruction between primary and each of the secondary SP-frames. One drawback of SP-frames is coding inefficiency. Due to lossless coding in secondary SP-frames, their sizes can be significantly larger than conventional P-frames. Furthermore, the number of secondary SP-frames required is equal to the number of SI frames, thus resulting in significant storage costs. As we will discuss, our proposed scheme encodes only one merge frame for all SI frames, and hence the storage requirement is lower than for SP-frames.

While DSC has been proposed for designing interactive and stream-switching mechanisms in the past decade [2, 5–7, 12], partly due to the computation complexity required for bit-plane and channel coding in common DSC implementations, DSC is not widely used nor adopted into any video coding standards. In contrast, in this work, our proposed coding tool involves only quantization (PWC function) and entropy coding of function parameters, both of which are computationally simple. Further, we demonstrate coding gain over a previously proposed DSC-based approach [7] in Section VII.

One of the primary applications of our proposed merge frame is interactive media systems, which have attracted considerable interest [13]. In particular, a range of media data types have been considered for interactive applications in the past: images [14], light-fields [15, 16], volumetric images [17], videos [5, 6, 18–22] and high-resolution videos [23–26]. While it is conceivable that our proposed merge frame can be applicable in some of these use scenarios for which DSC techniques have been proposed, here we focus on real-time switching among multiple pre-encoded video streams, as discussed in Section III.

This paper extends our earlier work [8], by providing a more detailed presentation and evaluation of the system, as well as introducing two new concepts. First, we study the fixed target merging case (Section V). Second, for the optimized target merging case, we develop a new algorithm to compute a locally optimal probability function $P(c)$ for shift c —one that leads to more efficient entropy coding of c , and small signal reconstruction distortion after merging (Section VI). We will show in our experiments, described in Section VII, that our new algorithm leads to significantly better RD performance than our previously published work [8].

III. SYSTEM OVERVIEW

A. IVSS System Overview

We provide an overview of our proposed coding system for *interactive video stream switching* (IVSS), in which our proposed *merge frame* is a key enabling component. In the sequel, a “picture” is a raw captured image in a video sequence, while a “frame” is a particular coded version of the picture (*e.g.*, I-frame, P-frame). In this terminology, a “picture” can have multiple coded versions or “frames”.

In an IVSS system, there are multiple pre-encoded video streams that are related (*e.g.*, videos capturing the same 3D scene from different viewpoints [3]). During video playback

¹An earlier version of this paper was presented at ICIP 2013 [8].

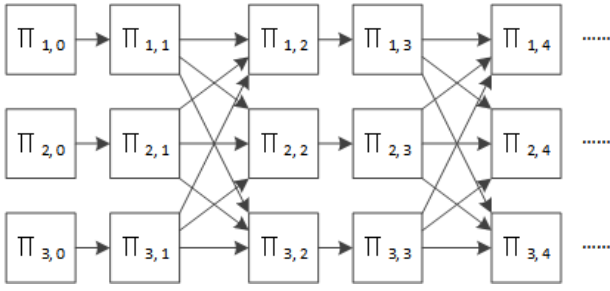


Fig. 2. Example of an acyclic picture interactivity graph for *dynamic view switching*. Each picture $\Pi_{v,t}$ has subscript indicating its view index v and time instant t . After viewing picture $\Pi_{2,1}$ of stream 2, the client can choose to keep watching the same stream and jump to $\Pi_{2,2}$, or switch to $\Pi_{1,2}$ or $\Pi_{3,2}$ of stream 1 and 3, respectively.

of a single stream, at a *switch instant*, the client can switch from a picture of the original stream to a picture of a different destination stream. Fig. 2 illustrates an example *picture interactivity graph* for three streams, where there is a switch instant every two pictures in time. An arrow $\Pi_p \rightarrow \Pi_q$ indicates that a switch is possible from picture Π_p to picture Π_q . This particular graph is *acyclic*, i.e., it has no loops and we cannot have both $\Pi_p \rightarrow \Pi_q$ and $\Pi_q \rightarrow \Pi_p$.

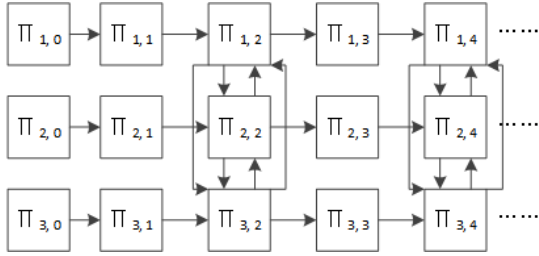


Fig. 3. Example of a cyclic picture interactivity graph for *static view switching*. Each picture $\Pi_{v,t}$ has subscript indicating its view index v and time instant t . After viewing $\Pi_{2,2}$ of stream 2, the client can choose to keep watching stream 2 in time and jump to $\Pi_{2,3}$, or change to $\Pi_{1,2}$ or $\Pi_{3,2}$ of stream 1 and 3, respectively, corresponding to the same time instant as $\Pi_{2,2}$.

The scenario in Fig. 2 is an example of *dynamic view switching* [27], where a frame at time t is always followed by a frame at time $t + 1$. In contrast, in *static view switching* a user can stop temporal playback and interactively select the angle from which to observe a 3D scene frozen in time [28]. Fig. 3 shows an example of static view switching, where the corresponding graph is *cyclic*, i.e., it contains loops so that we can have both $\Pi_p \rightarrow \Pi_q$ and $\Pi_q \rightarrow \Pi_p$. We will discuss the merge frame design for the cyclic case in Section V.

B. Stream-Switch Mechanism in IVSS

At a given switch instant, stream switching works as follows. First, for each possible switch $\Pi_p \rightarrow \Pi_q$, we encode a P-frame $P_{q|p}$ for Π_q , where a decoded version of Π_p is used as a predictor. Reconstructed $P_{q|p}$ is called a *side information* (SI) frame, which constitutes a particular reconstruction of destination Π_q . Because there are in general multiple origins for a given destination (the *in-degree* for destination picture in the picture interactivity graph), there are multiple corresponding SI frames. Having multiple reconstructions of the same picture

Π_q creates a problem for the following frame(s) that use Π_q as a predictor for predictive coding, because one does not know *a priori* which reconstructed SI frame $P_{q|p}$ will be available at the decoder buffer for prediction. This illustrates the need for our proposed merge frame (called *M-frame* in the sequel) M_q , which is an *extra* frame corresponding to destination Π_q . Correct decoding of M_q means a unique reconstruction of Π_q , no matter which SI frame $P_{q|p}$ is actually available at the decoder.

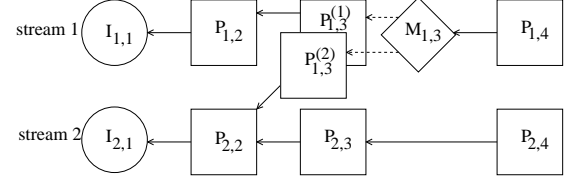


Fig. 4. Example of stream-switching from one pre-encoded stream to another using merge frame. SI frames $P_{1,3}^{(1)}$ and $P_{1,3}^{(2)}$ are first constructed using predictors $P_{1,2}$ and $P_{2,2}$, respectively. M-frame $M_{1,3}$ is encoded using the two SI frames. I-, P- and M-frames are represented as circles, squares and diamonds, respectively.

As an illustration, in Fig. 4 two P-frames, $P_{1,3}^{(1)}$ and $P_{1,3}^{(2)}$, generated from predictors $P_{1,2}$ and $P_{2,2}$ respectively, are the SI frames. An M-frame $M_{1,3}$ is added to merge the SI frames to produce an identical reconstruction for $\Pi_{1,3}$. During a stream-switch, the server can transmit any one of the two SI frames and $M_{1,3}$ leading to the same reconstructed frame for $\Pi_{1,3}$, thus avoiding coding drift in the following frame $P_{1,4}$. Note that one P-frame and one M-frame are sent. An alternative approach based on SP frames would require sending a primary SP-frame $S_{1,3}^1$ (using $P_{1,2}$ as the predictor) for the switch $\Pi_{1,2} \rightarrow \Pi_{1,3}$, or a losslessly coded secondary SP-frame $S_{1,3}^2$ (using $P_{2,2}$ as the predictor) for the switch $\Pi_{2,2} \rightarrow \Pi_{1,3}$. SP-frame approaches are asymmetric; rate is much lower when only a primary SP-frame is needed. In contrast, the switching cost using M-frame is always the same (P- and M-frames are transmitted). As will be shown, a combination of a P-frame and an M-frame requires lower rate than a secondary SP-frame.

C. Merge Frame Overview

In our proposed M-frame, each fixed-size code block in an SI frame is first transformed to the DCT domain. DCT coefficients are then quantized. The quantized coefficients across SI frames (called *q-coeffs* for short in the sequel) are then examined. If the q-coeffs of a given block are very different across SI frames, then the overhead to merge their differences to targeted q-coeffs would be large. Thus, we will encode the block as a conventional intra block. On the other hand, if the q-coeffs of a given block are already identical across all SI frames, then we can simply inform the decoder that the q-coeffs can be used without further processing. Finally, if the q-coeffs across SI frames are not identical but are similar, then each q-coeff is then merged identically to a target value via our proposed merge operator. Hence, together there are three coding modes for each code block: *intra*, *skip* and *merge*. In this paper, we focus our attention on optimizing

TABLE I
TABLE OF NOTATIONS

N	number of SI frames
\mathbf{S}^n	SI frame n
\mathbf{T}	desired target frame
\mathbf{M}	M-frame
$R(\mathbf{M})$	rate of M-frame \mathbf{M}
$D(\mathbf{T}, \bar{\mathbf{T}}(\mathbf{M}))$	distortion of reconstructed \mathbf{M} wrt \mathbf{T}
λ	weight parameter to trade off distortion with rate
\mathcal{B}_M	block group encoded in merge mode
K	number of pixels in a code block
\mathbf{x}_b^n	block b of SI frame \mathbf{S}^n
$Y_b^n(k)$	k -th DCT coefficient of block b of SI frame \mathbf{S}^n
$X_b^n(k)$	k -th q-coeff of block b of SI frame \mathbf{S}^n
Q	quantization step size
$X_b(k)$	k -th reconstructed q-coeff of block b
$Z_b(k)$	max. pair difference between any pair of $X_b^n(k)$
$Z_{\mathcal{B}_M}^*(k)$	group-wise max. pair difference, i.e. $\max_{b \in \mathcal{B}_M} Z_b^*(k)$
$W_{\mathcal{B}_M}(k)$	step size for k -th q-coeff of block group \mathcal{B}_M
$c_b(k)$	shift parameter for k -th q-coeff of block b
$\mathcal{F}_b(k)$	feasible range of shift c_b for identical merging
$Z_b(k)$	max. target diff. between target $X_b^0(k)$ and any $X_b^n(k)$
$Z_{\mathcal{B}_M}(k)$	group-wise max. target difference, i.e. $\max_{b \in \mathcal{B}_M} Z_b(k)$
$W_{\mathcal{B}_M}^\#(k)$	step size for k -th q-coeff for fixed target merging

the parameters in *merge* mode as the *intra* and *skip* modes are straightforward.

IV. PROBLEM FORMULATION

A. Notation

We first define the notation that will be used in the sequel; see Table I for quick reference. We denote the N SI frames by $\mathbf{S}^1, \dots, \mathbf{S}^N$, one of which is guaranteed to be available at the decoder buffer when M-frame \mathbf{M} is decoded. We denote a desired target picture by \mathbf{T} and for notational convenience we will include it in the set of SI frames as $\mathbf{S}^0 = \mathbf{T}$.

We denote the group of fixed-size code blocks in \mathbf{M} that are encoded in merge mode by \mathcal{B}_M . Each block has K pixels. We denote by \mathbf{x}_b^n the b -th block in SI frame \mathbf{S}^n coded in merge mode. Each block \mathbf{x}_b^n is transformed into the DCT domain as $\mathbf{Y}_b^n = [Y_b^n(0), \dots, Y_b^n(K-1)]$, where $Y_b^n(k)$ is the k -th DCT coefficient of \mathbf{x}_b^n . We denote by $X_b^n(k)$ the k -th quantized coefficient (*q-coeff*) given uniform quantization step size Q :

$$X_b^n(k) = \text{round}\left(\frac{Y_b^n(k)}{Q}\right), \quad (1)$$

where $\text{round}(x)$ is the standard rounding operation to the nearest integer.

B. Formulation

We consider two different problems based on the reconstruction requirement with respect to the desired target \mathbf{T} . One typically chooses \mathbf{T} *a priori*, e.g., by encoding the target picture independently (intra only) and using the decoded version as \mathbf{T} . The first problem requires the M-frame to reconstruct *identically* to desired target \mathbf{T} :

Problem 1. Fixed Target Merging (Section V). Find M-frame \mathbf{M} such that the decoder, taking as input any one of the SI frames \mathbf{S}^n and \mathbf{M} , can reconstruct \mathbf{T} identically as output.

Because of the differences between SI frames \mathbf{S}^n and desired target \mathbf{T} , there may be situations where a high rate is required for \mathbf{M} (e.g., due to motion in the video sequence, the target frame is very different from previously transmitted frames). In this case, we allow the reconstruction to deviate from desired target \mathbf{T} in order to reduce the rate required for \mathbf{M} by optimizing a rate-distortion criterion:

Problem 2. Optimized Target Merging (Section VI). Find \mathbf{M}^* and $\bar{\mathbf{T}}(\mathbf{M}^*)$ so that the decoder, taking as input any one of SI frames \mathbf{S}^n and \mathbf{M}^* , can always reconstruct $\bar{\mathbf{T}}(\mathbf{M}^*)$ as output, and where \mathbf{M}^* is an RD-optimal solution for a given weight parameter λ , i.e.,

$$\mathbf{M}^* = \arg \min_{\mathbf{M}} D(\mathbf{T}, \bar{\mathbf{T}}(\mathbf{M})) + \lambda R(\mathbf{M}), \quad (2)$$

where $D(\mathbf{T}, \bar{\mathbf{T}}(\mathbf{M}))$ is the distortion incurred (with respect to \mathbf{T}) when choosing $\bar{\mathbf{T}}(\mathbf{M})$ as the common reconstructed frame, and $R(\mathbf{M})$ is the rate needed to transmit \mathbf{M} .

The second problem essentially states that the *reconstruction target* $\bar{\mathbf{T}}(\mathbf{M})$ is RD-optimized with respect to desired target \mathbf{T} , while the first problem requires identical reconstruction to desired target \mathbf{T} . Note that in both problem formulations we avoid coding drift since they guarantee identical reconstruction for any SI frame, but a solution to Problem 2 will be shown to lead to significantly lower coding rates.

C. Piecewise Constant Function for Single Merging

A merge operation must, given q-coeff $X_b^n(k)$ of any SI frames \mathbf{S}^n , $n \in \{1, \dots, N\}$, reconstruct an identical value $\bar{X}_b(k)$, for all frequencies k . We use a PWC function $f(x)$ as the chosen merging operator, with *shift* c and *step size* W parameters selected for each frequency k of each block b encoded in merge mode (see Fig. 1). The selection of these parameters influences the RD performance of this merging operation for the optimized target merging case. We now focus our discussion on how c and W are selected for each coefficient. Because the optimization is the same for each frequency k , we will drop the frequency index k for simplicity of presentation.

Examples of PWC functions are ceiling, round, floor, etc. In this paper, we employ the floor function²:

$$f(x) = \left\lfloor \frac{x+c}{W} \right\rfloor W + \frac{W}{2} - c. \quad (3)$$

From Fig. 1, it is clear that there are numerous combinations of parameters W and c such that identical merging is ensured—i.e., all X_b^n map to the same constant interval. Note also that the choice of W depends on how spread out the various X_b^0, \dots, X_b^N are, that is, how correlated the SI blocks are to each other. In contrast, c is used to select a desired reconstruction value X_b^0 . Thus, because the level of correlation can be assumed to be relatively consistent across blocks, a *step size* $W_{\mathcal{B}_M}$ is selected once for all blocks $b \in \mathcal{B}_M$ for a given frequency. On the other hand, since the actual reconstruction

²We define floor function to minimize the maximum difference between original x and reconstructed $f(x)$, given shift c and step size W .

value will be different from block to block, the *shift* c_b will be selected on a per block basis for a given frequency.

Before formulating the problem of optimizing the choice of c and W , we derive constraints under which this selection is made by determining:

- The minimum value of W that guarantees identical merging,
- The choice of c that guarantees correct reconstruction,
- Effective range of c .

We first compute a minimum step size W to enable identical merging for blocks b in \mathcal{B}_M . Let Z_b^* be the *maximum pair difference* between any pair of q-coeffs of a given frequency in block b , i.e.,

$$Z_b^* = \max_{i,j \in \{0, \dots, N\}} X_b^i - X_b^j = X_b^{\max} - X_b^{\min}, \quad (4)$$

where X_b^{\max} and X_b^{\min} are respectively the maximum and minimum q-coeffs among the SI frames, i.e.,

$$X_b^{\max} = \max_{n=0, \dots, N} X_b^n, \quad X_b^{\min} = \min_{n=0, \dots, N} X_b^n. \quad (5)$$

Given Z_b^* , we next define the *group-wise maximum pair difference* $Z_{\mathcal{B}_M}^*$ for the blocks in group \mathcal{B}_M :

$$Z_{\mathcal{B}_M}^* = \max_{b \in \mathcal{B}_M} Z_b^*. \quad (6)$$

Since all X_b^n are integer, $Z_{\mathcal{B}_M}^*$ is also an integer. We can now establish a minimum for step size $W_{\mathcal{B}_M}$ above which identical merging for all blocks $b \in \mathcal{B}_M$ is achievable:

Fact 1. Minimum Step Size for Identical Merging: a step size $W_{\mathcal{B}_M} > Z_{\mathcal{B}_M}^*$, is large enough for floor function $f(X_b^n)$ in (3) to merge any X_b^n in \mathcal{B}_M to a same value \bar{X}_b .

Since each \mathbf{S}^n is a coarse approximation of (and thus is similar to) desired target \mathbf{T} , the \mathbf{S}^n 's themselves are similar. Hence, the largest difference Z_b^* should be small in the typical case. Indeed, we observe empirically that Z_b^* follows an exponential distribution (one-sided because Z_b^* is non-negative). Fig. 5 shows Z_b^* probability distribution for $k = 16$ and $k = 32$. We can see that 80% of the blocks have $Z_b^* \leq 5$. Assuming that Z_b^* follows a Laplacian distribution, the maximum $Z_{\mathcal{B}_M}^*$ is typically much larger than the average Z_b^* . This will be shown to be useful for the optimized merging of Section VI.

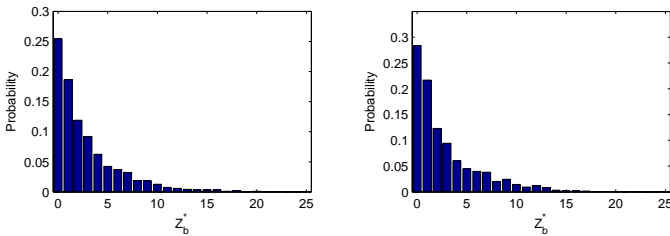


Fig. 5. Two examples of probability distribution of Z_b^* with three SI frames at $Q = 1$ for Balloons at frequency $k = 16$ and $k = 32$.

Fact 1 states that step size $W_{\mathcal{B}_M}$ is wide enough so that X_b^0, \dots, X_b^N can all fall on the same interval in $f(x)$, as shown in Fig. 1. However, given $W_{\mathcal{B}_M}$, shift c_b must still be appropriately chosen *per block* to achieve identical merging.

Mathematically, identical merging means that the floor function with parameters c_b and $W_{\mathcal{B}_M}$ produces the same integer output for all inputs X_b^n , that is:

$$\left\lfloor \frac{X_b^n + c_b}{W_{\mathcal{B}_M}} \right\rfloor = \left\lfloor \frac{X_b^0 + c_b}{W_{\mathcal{B}_M}} \right\rfloor, \quad \forall n \in \{1, \dots, N\}. \quad (7)$$

Thus for all X_b^n , we must have for some $m \in \mathbb{Z}$ that:

$$mW_{\mathcal{B}_M} \leq X_b^n + c_b < (m+1)W_{\mathcal{B}_M}, \quad \forall n \in \{0, \dots, N\} \quad (8)$$

Instead of considering all X_b^n 's, it is sufficient to consider only the maximum and minimum values, so that the maximum range for c_b that guarantees identical reconstruction is:

$$mW_{\mathcal{B}_M} - X_b^{\min} \leq c_b < (m+1)W_{\mathcal{B}_M} - X_b^{\max} \quad (9)$$

for some integer m . Note that given step size $W_{\mathcal{B}_M}$, c_b and $c_b + mW_{\mathcal{B}_M}$ lead to the same output:

$$\begin{aligned} f(x) &= \left\lfloor \frac{x + c_b + mW_{\mathcal{B}_M}}{W_{\mathcal{B}_M}} \right\rfloor W_{\mathcal{B}_M} + \frac{W_{\mathcal{B}_M}}{2} - (c_b + mW_{\mathcal{B}_M}) \\ &= \left\lfloor \frac{x + c_b}{W_{\mathcal{B}_M}} \right\rfloor W_{\mathcal{B}_M} + \frac{W_{\mathcal{B}_M}}{2} - c_b \end{aligned}$$

Thus it will be sufficient to consider at most W different values of c_b as possible candidates.

Define $\alpha = X_b^{\min} \bmod W_{\mathcal{B}_M}$ and $\beta = X_b^{\max} \bmod W_{\mathcal{B}_M}$ and consider the two possible cases.

- In case (i) $X_b^{\min} = mW_{\mathcal{B}_M} + \alpha$ and $X_b^{\max} = mW_{\mathcal{B}_M} + \beta$, where $\alpha < \beta$, so that X_b^{\min} and X_b^{\max} fall in the same interval when there is no shift, $c_b = 0$. Hence we can have $-\alpha \leq c_b < W_{\mathcal{B}_M} - \beta$ in order to keep both X_b^{\min} and X_b^{\max} in the interval $[mW_{\mathcal{B}_M}, (m+1)W_{\mathcal{B}_M})$.
- In case (ii) $X_b^{\min} = mW_{\mathcal{B}_M} + \alpha$ and $X_b^{\max} = (m+1)W_{\mathcal{B}_M} + \beta$, where $\beta < \alpha$, i.e., when $c_b = 0$, X_b^{\min} and X_b^{\max} fall in neighboring intervals. Here we can have $-\alpha \leq c_b < -\beta$ to move X_b^{\max} down to the interval $[mW_{\mathcal{B}_M}, (m+1)W_{\mathcal{B}_M})$, or have $W_{\mathcal{B}_M} - \alpha < c_b \leq W_{\mathcal{B}_M} - \beta$ to move X_b^{\min} up to the interval $[(m+1)W_{\mathcal{B}_M}, (m+2)W_{\mathcal{B}_M})$.

Note that the selection of $W_{\mathcal{B}_M}$ (Fact 1) implies that $X_b^{\max} - X_b^{\min} < W_{\mathcal{B}_M}$, and $\alpha = \beta$ only if $X_b^{\min} = X_b^{\max}$, in which case there is no merging needed and any c_b would suffice.

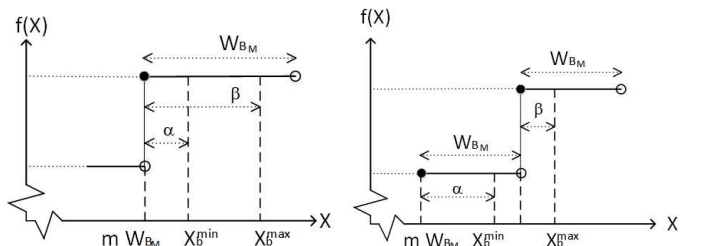


Fig. 6. Two cases of X_b^{\min} and X_b^{\max} (left: $\alpha < \beta$ and right: $\alpha > \beta$) and their implications on the feasible range of shift c_b .

The two cases ($\alpha < \beta$ and $\alpha > \beta$) are illustrated in Fig. 6. Note that given $X_b^{\max} \geq X_b^{\min}$ by definition, we will be in Case (ii) whenever $\beta < \alpha$. Thus we can summarize this result as:

Fact 2. Maximum Feasible Range \mathcal{F}_b for Shift c_b : For the shift c_b to provide identical merging of q-coeffs X_b^0, \dots, X_b^N to

a same value \bar{X}_b , given step size $W_{\mathcal{B}_M}$

$$c_b \in \mathcal{F}_b = [-\alpha, W_{\mathcal{B}_M} - \beta] \text{ if } \alpha < \beta$$

and

$$c_b \in \mathcal{F}_b = [W_{\mathcal{B}_M} - \alpha, W_{\mathcal{B}_M} - \beta] \text{ if } \alpha > \beta$$

with $\alpha = X_b^{\min} \bmod W_{\mathcal{B}_M}$ and $\beta = X_b^{\max} \bmod W_{\mathcal{B}_M}$.

D. Formulation of Merge Frame RD-Optimization

In order to formulate the PWC function parameter optimization problem, we first define distortion, d_b , as the squared difference between coefficient Y_b^0 of the desired target \mathbf{T} and reconstructed coefficient $f(X_b^0)Q$:

$$d_b = |Y_b^0 - f(X_b^0)Q|^2. \quad (10)$$

Because shift c_b will be always chosen within the feasible range defined in Fact 2, all q-coeffs X_b^n will map to the same value $f(X_b^n)$, $\forall n \in \{0, \dots, N\}$. Thus we only need to compute the distortion for $f(X_b^0)$ in (10).

For the k -th q-coeff in block group \mathcal{B}_M , the encoder will have to transmit to the decoder:

- 1) one step size $W_{\mathcal{B}_M}(k) > Z_{\mathcal{B}_M}(k)$ for each group \mathcal{B}_M .
- 2) one shift $c_b(k)$ for each block b in group \mathcal{B}_M .

The cost of encoding a single $W_{\mathcal{B}_M}(k)$ for all k -th q-coeffs in group \mathcal{B}_M is small, while the cost of encoding $|c_b(k)|$ shifts $c_b(k)$ for each of the k -th q-coeffs can be significant. Thus we consider only the rate associated to $c_b(k)$ in our optimization.

Note that since the high-frequency DCT coefficients of a given code block are very likely zero, we can insert an *End of Block* (EOB) flag E_b to signal the remaining high-frequency q-coeffs in block b in a raster-scan order are 0. Effective use of E_b can reduce the amount of transmitted PWC function parameters³. In summary, we can define the RD optimized target merging problem as:

$$\min_{W_{\mathcal{B}_M}(k), c_b(k)} \sum_{b \in \mathcal{B}_M} D_b + \lambda R_b, \quad \begin{matrix} W_{\mathcal{B}_M}(k) > Z_{\mathcal{B}_M}(k) \\ c_b(k) \in \mathcal{F}_b(k) \end{matrix} \quad (11)$$

with distortion D_b and rate R_b for block b calculated as:

$$\begin{aligned} D_b &= \sum_{k=0}^{E_b} d_b(k) + \sum_{k=E_b+1}^{K-1} Y_b^0(k)^2 \\ R_b &= \sum_{k=0}^{E_b} R(c_b(k)), \end{aligned}$$

where $d_b(k)$ is defined in (10) and $R(c_b(k))$ is the rate to encode $c_b(k)$. We discuss how we tackle this optimization in Section VI.

V. FIXED TARGET MERGING

In certain applications, such as the static view switching scenario discussed in Section III and illustrated in Fig. 3, the picture interactivity graph is cyclic, so that we may have that

³In the fixed target merging case, E_b is inserted when the remaining high-frequency q-coeffs of a block b in target \mathbf{T} are exactly zero. In the optimized target case, E_b can be inserted in an RD-optimal manner on a per-block basis, similar to what is done in coding standards such as H.264 [11].

$\Pi_p \rightarrow \Pi_q$ and $\Pi_q \rightarrow \Pi_p$. Because of this interdependency, one cannot directly define a simple target merging optimization, since optimizing the reconstruction for Π_q would require first fixing a representation (frame) for Π_p , but optimizing Π_p would in turn require first fixing a representation for Π_q . As a simple alternative we propose *fixed target merging*, where the reconstruction target \mathbf{T} for each picture is chosen independently from the SI frames. For example, \mathbf{T} can be the I-frame of the target picture for a given QP.

A. Fixed Target Reconstruction using Merge Operator

We first show that given a target reconstruction value a and a step size W , we can always find a shift c so that $f(x)$ in (3) is such that $f(x) = a$ for all inputs x in the interval $[a - W/2, a + W/2)$. To see this, first write target reconstruction value $a = a_1W + a_2$, where a_1 and $a_2 = a \bmod W$ are integers and $0 \leq a_2 < W$. Similarly, we write input $x = a_1W + x_2$ where integer x_2 can be bounded:

$$\begin{aligned} a - \frac{W}{2} &\leq x < a + \frac{W}{2} \\ a_1W + a_2 - \frac{W}{2} &\leq a_1W + x_2 < a_1W + a_2 + \frac{W}{2} \\ a_2 - \frac{W}{2} &\leq x_2 < a_2 + \frac{W}{2} \end{aligned} \quad (12)$$

We now set $c = \frac{W}{2} - a_2$. We show that this ensures $f(x) = a$ for $x \in [a - W/2, a + W/2)$:

$$\begin{aligned} f(x) &= \left\lfloor \frac{a_1W + x_2 + \frac{W}{2} - a_2}{W} \right\rfloor W + \frac{W}{2} - \left(\frac{W}{2} - a_2 \right) \\ &= a_1W + a_2 = a. \end{aligned} \quad (13)$$

where the second line is true because $x_2 + \frac{W}{2} - a_2$ in the numerator of the “round-down” operator argument can be bounded in $[0, W)$ using (12):

$$\begin{aligned} a_2 - \frac{W}{2} + \frac{W}{2} - a_2 &\leq x_2 + \frac{W}{2} - a_2 < a_2 + \frac{W}{2} + \frac{W}{2} - a_2 \\ 0 &\leq x_2 + \frac{W}{2} - a_2 < W \end{aligned} \quad (14)$$

Next, recall from Section IV-C that we include the desired target \mathbf{T} as the first SI frame \mathbf{S}^0 . For a given frequency of a particular block b , we first compute the *maximum target difference* Z_b as the largest absolute difference between target q-coeff X_b^0 and X_b^n of any SI frame \mathbf{S}^n , i.e.,

$$Z_b = \max_{n \in \{1, \dots, N\}} |X_b^0 - X_b^n| \quad (15)$$

Based on this we can choose step size and shift based on the following lemma.

Lemma V.1. *Choosing step size $W_b^\# = 2Z_b + 2$ and shift $c_b = W_b^\#/2 - X_{b,2}^0$, where $X_{b,2}^0 = X_b^0 \bmod W_b^\#$, guarantees that $f(X_b^n) = X_b^n$, $\forall n \in \{0, \dots, N\}$. Note that $W_b^\#$ is an even number, and c is an integer as required.*

Proof: Given shift $c_b = W_b^\#/2 - X_{b,2}^0$, showing $X_b^n \in [X_b^0 - W_b^\#/2, X_b^0 + W_b^\#/2)$ implies $f(X_b^n) = X_b^n$, $\forall n \in \{0, \dots, N\}$. Defining step size $W_b^\# = 2Z_b + 2$ means the required interval for X_b^n can be rewritten as $[X_b^0 - Z_b - 1, X_b^0 + Z_b + 1)$. By the

definition of Z_b , we know $X_b^0 - Z_b \leq X_b^n \leq X_b^0 + Z_b$. Hence the required interval for X_b^n is met. ■

Note that we can achieve fixed target merging for a given X_b^0 as long as the step size is larger than $W_b^\#$. For example, we can assign the same step size $W_{\mathcal{B}_M}^\#$ for all blocks in a group \mathcal{B}_M , so that we reduce the rate overhead:

$$W_{\mathcal{B}_M}^\# = 2 + 2Z_{\mathcal{B}_M} \quad (16)$$

where $Z_{\mathcal{B}_M} = \max_{b \in \mathcal{B}_M} Z_b$ is the *group-wise maximum target difference*, and Z_b , the block-wise maximum target difference for block b , is computed using (15). In summary:

- 1) We define a set of blocks \mathcal{B}_M and use $W_{\mathcal{B}_M}^\#(k)$ computed using (16) for frequency k of all blocks in \mathcal{B}_M .
- 2) For block b , we set shift $c_b(k) = W_{\mathcal{B}_M}^\#(k)/2 - X_{b,2}^0(k)$, where $X_{b,2}^0(k) = X_b^0(k) \bmod W_{\mathcal{B}_M}^\#(k)$. A different shift is used for each frequency k and block b , and transmitted as part of the M-frame along with $W_{\mathcal{B}_M}^\#(k)$.

VI. OPTIMIZED TARGET MERGING

We now propose a merging approach based on selecting $W_{\mathcal{B}_M}(k)$ and $c_b(k)$ so as to find a solution to the optimization problem described in Section IV-D, where we allow the reconstructed value to be different from $X_b^0(k)$.

If $W_{\mathcal{B}_M}$ is chosen large enough, *i.e.* $W_{\mathcal{B}_M} \geq 2 + 2Z_{\mathcal{B}_M}$, then we have shown (Lemma V.1) that one can select shift c_b to reconstruct target q-coeff X_b^0 exactly. However, the shifts are a function of $X_{b,2}^0 = X_b^0 \bmod W_{\mathcal{B}_M}$ (Lemma V.1), and thus we can expect them to have a uniform distribution, which would mean that a rate of the order of $\log_2(W_{\mathcal{B}_M})$ would be required as overhead. In order to reduce this rate, we use two approaches: i) we allow $W_{\mathcal{B}_M}$ to be smaller than required by Lemma V.1, and ii) when multiple choices of c_b provide identical reconstruction, we optimize this choice based on the criteria introduced in Section IV-D.

A. Selection of $W_{\mathcal{B}_M}$

Note, by definition of $Z_{\mathcal{B}_M}^*$, we are guaranteed that all X_b^n can be within an interval of size $W_{\mathcal{B}_M}$ as long as $W_{\mathcal{B}_M} > Z_{\mathcal{B}_M}^*$, provided we transmit an appropriate c_b (Fact 1). Reducing $W_{\mathcal{B}_M}$ from $2 + 2Z_{\mathcal{B}_M}$ can reduce the rate required to transmit c_b , since c_b can take at most $W_{\mathcal{B}_M}$ different values.

As shown in Section IV-C we observe empirically that Z_b^* follows a Laplacian distribution (Fig. 5). Thus, for a large block group \mathcal{B}_M , $Z_{\mathcal{B}_M}^* = \max_{b \in \mathcal{B}_M} Z_b^*$ will be in general much larger than Z_b^* . Since $Z_b^* \geq Z_b$, in practice for many blocks b it is thus possible to reconstruct target X_b^0 since $W_{\mathcal{B}_M}^* > 2Z_b + 2$. Thus, we propose to select $W_{\mathcal{B}_M} = Z_{\mathcal{B}_M}^* + 1$, which guarantees that for the worst case block all SI values are in the same interval, with appropriate choice of c_b to be discussed next.

B. RD-optimal Selection of Shifts

Given a chosen $W_{\mathcal{B}_M}$, according to Fact 2 there will be multiple values of c_b that guarantee identical reconstruction for all X_b^n . To enable efficient entropy coding of c_b , it is desirable to have a skewed probability distribution $P(c_b)$ of c_b . We design an algorithm to promote a skewed $P(c_b)$ iteratively. We first

propose how to initialize $P(c_b)$, and then discuss how to update $P(c_b)$ in subsequent iterations.

We optimize shift c_b via the following RD cost function:

$$\min_{0 \leq c_b < W_{\mathcal{B}_M} \mid c_b \in \mathcal{F}_b} d_b + \lambda(-\log P(c_b)), \quad (17)$$

where the rate term is approximated as the negative log of the probability $P(c_b)$ of candidate c_b , and d_b is the distortion term computed using (10). The difficulty in using objective (17) to compute optimal c_b^* lies in how to define $P(c_b)$ prior to selection of c_b . Our strategy is to initialize a skewed distribution $P(c_b)$ to promote a low coding rate, perform optimization (17) for each block $b \in \mathcal{B}_M$, then update $P(c_b)$ based on statistics of the selected c_b 's, and repeat until $P(c_b)$ converges.

In order to choose an initial distribution $P(c_b)$, we note that a distribution with a small number of spikes has lower entropy than a smooth distribution (see Fig. 7 as an example). Choosing c_b values following such a discrete distribution (*e.g.*, left in Fig. 7) means that we reduce the number of possible c_b , which may increase d_b . Thus, if λ in (17) is small, in order to reduce distortion one can increase the number of spikes in $P(c_b)$. In this paper, we propose to induce a multi-spike probability $P(c_b)$, where the appropriate number of spikes depends on the desired tradeoff between distortion and rate in (17).

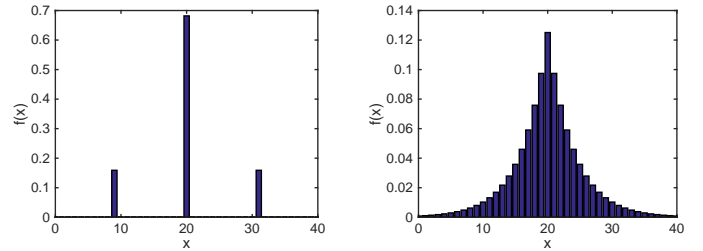


Fig. 7. Two examples of shift distribution $P(c_b)$. Left distribution has small number of spikes and has low entropy (1.22). Right distribution is smooth but has high entropy (4.38).

Since c_b is constrained to be in the feasible region \mathcal{F}_b defined in Fact 2, it is possible that when we restrict c_b to just a few values as in Fig. 7 (left), there will be some blocks b for which none of the “spikes” in $P(c_b)$ fall within their \mathcal{F}_b . In order to guarantee identical reconstruction they must be able to select non-spike values as shifts c_b . Thus we propose a “spike + uniform” distribution $P(c_b)$:

$$P(c_b) = \begin{cases} p_i^s & \text{if } c_b = c_i^s \\ p_c & \text{o.w.} \end{cases} \quad (18)$$

where $\{c_1^s, \dots, c_H^s\}$ are the H spikes, each with probability p_i^s , and p_c is a small constant for non-spike shift values. p_c is chosen so that $P(c_b)$ sums to 1.

1) *Computing distribution $P(c_b)$ for fixed H :* We now discuss how we compute $P(c_b)$ for given H . Empirically we observe that for a reasonable number of spikes (*e.g.*, $H \geq 3$), the majority of blocks (typically 99% or more) in \mathcal{B}_M have at least one spike in their feasible region \mathcal{F}_b . Thus, to simplify our computation we first ignore the feasibility constraint and

employ an iterative *rate-constrained Lloyd-Max* algorithm (rc-LM) [29] to identify spike locations.

We illustrate the operations of rc-LM to initialize H spike locations for $H = 3$ as follows. Let c_b^o be the shift value that minimizes *only* distortion for block b . Let $g(c^o)$ be the probability distribution of distortion-minimizing shift c^o for blocks in \mathcal{B}_m , where $0 \leq c^o < W_{\mathcal{B}_M}$. $g(c^o)$ can be computed empirically for group \mathcal{B}_m . Without loss of generality, we define quantization bins for the three spikes c_1^s , c_2^s and c_3^s as $[0, b_1)$, $[b_1, b_2)$ and $[b_2, W_{\mathcal{B}_M})$ respectively. The expected distortion $D(\{c_i^s\})$ given three spikes is:

$$\sum_{c^o=0}^{b_1-1} |c^o - c_1^s|^2 g(c^o) + \sum_{c^o=b_1}^{b_2-1} |c^o - c_2^s|^2 g(c^o) + \sum_{c^o=b_2}^{W_{\mathcal{B}_M}-1} |c^o - c_3^s|^2 g(c^o) \quad (19)$$

where $D(\{c_i^s\})$ is computed as the sum of squared difference between c^o and spike c_i^s in the bin that c^o is assigned to. Having defined distortion $D(\{c_i^s\})$, the initial spike locations c_i^s given H spikes can be found as follows: i) construct H spikes evenly spaced in the interval $[0, W_{\mathcal{B}_M})$, ii) use conventional Lloyd-Max algorithm with no rate constraints to converge to a set of H bin centroids c_i^s .

Next, adding consideration for rate, the RD cost of the three spikes can then be written as:

$$D(\{c_i^s\}) + \lambda \left(-\log\left(\sum_{c^o=0}^{b_1-1} g(c^o)\right) - \log\left(\sum_{c^o=b_1}^{b_2-1} g(c^o)\right) - \log\left(\sum_{c^o=b_2}^{W_{\mathcal{B}_M}-1} g(c^o)\right) \right) \quad (20)$$

(20) is essentially the aggregate of RD costs (17) for all blocks in \mathcal{B}_M .

To minimize (20), rc-LM alternately optimizes bin boundaries b_i and spike locations c_i^s at a time until convergence. Given spikes c_i^s are fixed, each bin boundary b_i is optimized via exhaustive search in the range $[c_i^s, c_{i+1}^s)$ to minimize both rate and distortion in (20). Given bin boundaries b_i are fixed, optimal c_i^s can be computed simply as the bin average:

$$c_i^s = \frac{\sum_{c^o=b_i}^{b_{i+1}-1} g(c^o) c^o}{\sum_{c^o=b_i}^{b_{i+1}-1} g(c^o)} \quad (21)$$

where $b_0 = 0$ and $b_3 = W_{\mathcal{B}_M}$.

Upon convergence, we can then identify the small fraction of blocks with no spikes in their feasible regions \mathcal{F}_b and assign an appropriate constant p_c so that $P(c_b)$ is well defined according to (18). Computing $P(c_b)$ with H spikes where $H \neq 3$ can be done similarly.

2) *Finding the optimal $P(c_b)$* : To find the optimal $P(c_b)$, we add an outer loop for this $P(c_b)$ construction procedure to search for the optimal number of spikes H . Pseudo-code of the complete algorithm is shown in Algorithm 1. We note that in practice, we observe that the number of iterations until convergence is small.

C. Comparison with Coset Coding

We now discuss the similarity between our proposed approaches and coset coding methods in DSC [9]. Consider first fixed target merging of one q-coeff of a single block b . In a scalar implementation of coset coding, given possible SI values

Algorithm 1 Computing the optimal shift distribution $P(c_b)$

```

1: for each number of spikes  $H \in [1, W_{\mathcal{B}_M}]$  do
2:   Initialize distribution  $P^0(c_b)$  via LM;
3:    $t = 0$ ;
4:   repeat
5:      $t = t + 1$ ;
6:     Update  $H$  spike locations  $c_i^s$  via (21);
7:     Update bin boundaries  $b_i$  by minimizing (20);
8:     Compute  $p_c$  for a new  $P^t(c_b)$ ;
9:   until  $\|P^{t-1}(c_b) - P^t(c_b)\| \leq \epsilon$ 
10: end for

```

$X_b^n, n \in \{1, \dots, N\}$, seen as “noisy” versions of a target X_b^0 , the largest difference $Z_b = \max_n |X_b^n - X_b^0|$ with respect to X_b^0 is first computed. The size of the coset W is then selected such that $W > 2Z_b$. The coset index $i_b = X_b^0 \bmod W$ is computed at the encoder for transmission.

At the decoder, the reconstructed value \hat{X}_b is the integer closest to received SI X_b^n with the same coset index i_b , i.e.,

$$\hat{X}_b = \arg \min_{X \in \mathbb{Z}} |X_b^n - X| \quad \text{s.t. } i_b = X \bmod W \quad (22)$$

Using the aforementioned coset coding scheme for blocks $b \in \mathcal{B}_M$, coding of $i_b = X_b^0 \bmod W = X_{b,2}^0$ per block is necessary, where coset size W is chosen such that $W > 2Z_{\mathcal{B}_M}$. In our fixed target merging scheme using PWC functions, we code a shift $c_b = W_{\mathcal{B}_M}^\# / 2 - X_{b,2}^0$ for each block b , where step size $W_{\mathcal{B}_M}^\#$ is also proportional to $2Z_{\mathcal{B}_M}$. Comparing the two schemes one can see that the number of choices that need to be sent to the decoder is the same (one of $W_{\mathcal{B}_M}^\#$ possible values in both cases). Both the shift value c_b and i_b are functions of $X_{b,2}^0$, the LSBs of X_b^0 , which are likely to have an approximately uniform distribution. Thus so the overhead rate should be the same for both coset coding and fixed target merging.

Consider now the optimized merging case. In this scenario we are able to choose $W_{\mathcal{B}_M} = Z_{\mathcal{B}_M}^* + 1$ —likely much smaller than $2Z_{\mathcal{B}_M} \leq 2Z_{\mathcal{B}_M}^*$ —so that we can still guarantee identical reconstruction, with a reduction in rate that comes at the cost of an increase in distortion. As for the coset coding approach, if we were to reduce to choose a smaller $W_{\mathcal{B}_M}$ as well, we in fact can no longer guarantee identical reconstruction. This is because when $W_{\mathcal{B}_M} < 2Z_{\mathcal{B}_M}$ there will be cases where not all the X_b^n are in the same interval, and thus the same i_b will lead to two different values at the decoder depending on the SI received. This imperfect merging will lead to undesirable coding drift in the following predicted frames, as discussed in Section III.

VII. EXPERIMENTS

We first discuss the general experimental setup and M-frame parameter selection (Section VII-A). We then verify the effectiveness of our proposed “Spike + Uniform” distribution (Section VII-B). Next, we compare the performance of our M-frame in three different situations: 1) static view switching (*Scenario 1* in Section VII-C); 2) switching among streams of different rates for the same single-view video (*Scenario 2*

in Section VII-D), and 3) dynamic view switching of multi-view videos of different viewpoints and encoded in the same bit-rate (*Scenario 3* in Section VII-E).

A. Experimental Setup

We use four different multiview video test sequences with resolution 1024x768 for scenarios 1 and 3: Balloons, Kendo⁴, Lovebird1 and Newspaper⁵. The viewpoints of each sequence are shown in Table II. For scenario 2, we use four single-view video sequences with resolution 1920x1080: BasketballDrive, Cactus, Kimono1 and ParkScene⁶.

TABLE II
VIEWPOINTS OF EACH MULTIVIEW SEQUENCES.

Sequence Name	Viewpoints
Balloons	1, 3, 5
Kendo	1, 3, 5
Lovebird1	4, 6, 8
Newspaper	3, 4, 5

We compare the coding performance of our proposed scheme against two schemes⁷: SP-frame [10] in H.264 and D-frame proposed in [30]. QP for D-frame is set to be equal to QP_{SI} to maintain consistent quality. For multi-view scenarios 1 and 3, we encoded three streams from three viewpoints: the center view was set as the target, to which the other two side views can switch at a defined switching point. For Scenario 2, we encoded the single-view video in three different bit-rates and then switched among them. The bit-rates for the three streams were decided according to *additive increase multiple decrease* (AIMD) rate control behavior in TCP and TFRC [31]: one stream has twice the target stream's bit-rate, while the other has slightly smaller bit-rate (0.9 times of the target stream's bit-rate). The results are shown in plots of PSNR versus coding rate for a switched frame.

M-frame parameters are selected as follows. In Scenario 1, different QP_M will result in different rates, and so we set QP_M to equal to QP_{SI} , as was done for D-frames. However, for optimized target merging, coding rate is determined mainly by the number of spikes in the distribution, and not QP_M . In our experiments, as similarly done in High Efficiency Video Coding (HEVC), we first empirically compute λ as a function of the SI frame's QP_{SI} :

$$\lambda = 2^{0.6QP_{SI}-12}, \quad (23)$$

The number of spikes in the distribution is driven by the selected λ . We then set $QP_M = 1$ to maintain small quantization error. For mode selection among *skip*, *intra* and *merge*, for each block b we first examine q-coeffs $X_b^n(k)$ of N SI frames. If $X_b^n(k)$ of all K frequencies are identical across the SI frames,

then block b is coded as *skip*. Otherwise, selection between *intra* and *merge* is done based on a RD criteria.

In HEVC, large code block sizes are introduced which bring significant coding gain on high resolution sequences [32]. Motivated by this observation, we also investigated the effect of different block sizes (4×4 , 8×8 , 16×16) on coding performance. We also compare our current proposal against the performance of our previous work [8], where block size is fixed at 8×8 , initial probability distribution of shift $P(c_b)$ is not optimized, and no RD-optimized EOB flag is employed. The corresponding PSNR-bitrate curves for scenario 3 are shown in Fig. 8.

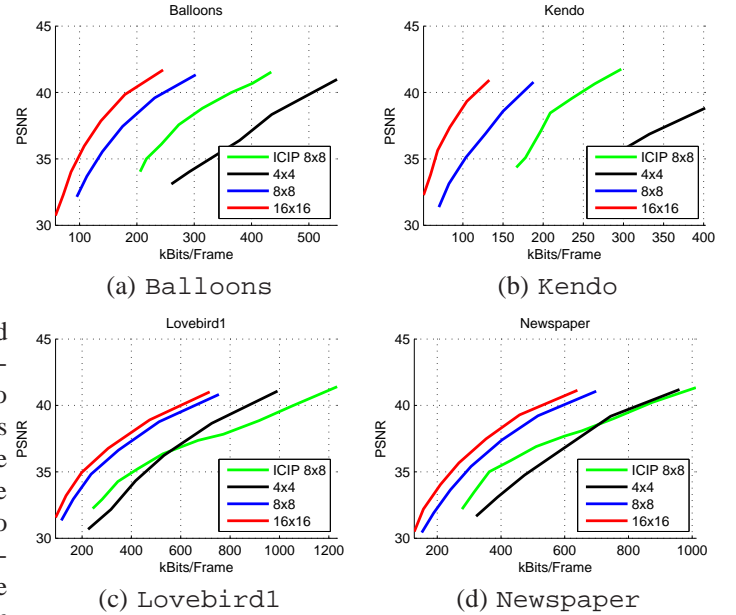


Fig. 8. PSNR v.s. encoding rate comparison with different block sizes for sequences Balloons, Kendo, Lovebird1 and Newspaper.

From Fig. 8, we observe that block size 16×16 provides the best coding performance at all bit-rates. One reason for the superior performance of large blocks in M-frame is the following: because SI frames are already reconstructions of the target frames (albeit slightly different), motion compensation is not necessary, so the benefit of smaller blocks typical in video coding is diminished. We note that in general an optimal block size per frame can be selected by the encoder *a priori* and encoded as side information to inform the decoder. In the following experiments, the block size will be fixed at 16×16 for best performance.

Further, we observe also that our proposed method achieves a significant coding performance gain compared to our previous method in [8] over all bit-rate regions, showing the effectiveness of our newly proposed optimization techniques.

B. Effectiveness of “Spike + Uniform” Distribution

In order to verify the effectiveness of our proposed “Spike + Uniform” (SpU) probability distribution $P(c_b)$ for shift parameter c_b , we choose a competing naïve distribution for $P(c_b)$ as follows: first, we compute distortion-minimizing $g(c^0)$ as the initial probability distribution. Next, we compute the RD-optimal c_b for each block $b \in \mathcal{B}_M$ via (17) for a single iteration

⁴http://www.tanimoto.nuee.nagoya-u.ac.jp/mpeg/mpeg_ftv.html

⁵<ftp://203.253.128.142>

⁶<ftp://ftp.tnt.uni-hannover.de/testsequences/>

⁷Here QP_A denotes the quantization parameter for coding DCT coefficients in approach A

using the initialized probability distribution and compute a new $P'(c_b)$. This $P'(c_b)$ is then used to compute the rate to encode each c_b of a merge block b . The difference between $P'(c_b)$ and our proposed $P^l(c_b)$ is that $P'(c_b)$ in general is an arbitrarily shaped distribution, not a skewed “spiky” distribution. Experimental results of M-frame using these distributions are shown in Fig. 9.

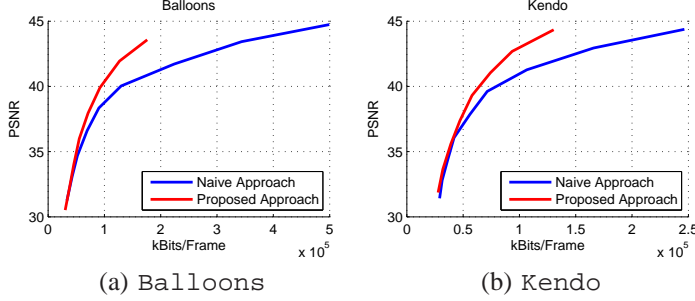


Fig. 9. PSNR v.s. encoding rate comparison with different block sizes for sequences Balloons, Kendo.

We observe from Fig. 9 that our proposed SpU distribution outperforms the naïve distribution in the high bit-rate region and is comparable in the low bit-rate region. This is because in the low bit-rate region λ is very large, so that for any initial distribution, after one iteration, there will only remain one spike, and the number of iterations required for convergence is very small.

C. Scenario 1: Static View Switching

We first test our proposed M-frame in the static view switching scenario for multi-view sequences. Three views are encoded using same QP. The fixed target merging algorithm described in Section V is used to facilitate switching to neighboring views among pictures of the same instant, as shown in Fig. 3.

Specifically, we constructed M- / D- frames to enable static view-switching from view 1 or 3 to target view 2. We first use H.264 to encode two SI frames (P-frames) using $\Pi_{2,2}$ as the target and $\Pi_{1,2}$ and $\Pi_{3,2}$ as predictors, respectively. This results in encoded rates $\mathcal{R}_{1,2}$ and $\mathcal{R}_{2,2}$ for the two SI frames, respectively. Then we encoded a M- / D- frame to merge these two SI frames identically to $\Pi_{2,2}$. The corresponding rates for M-frame and D-frame are $\mathcal{R}_{2,2}^M$ and $\mathcal{R}_{2,2}^D$, respectively. Since SP-frame in H.264 cannot perform fixed target merging, it is not tested in this scenario.

We assume that the switching probability is equal on both view 1 and 3, which is 0.5. Then the overall rate for the D-frame is calculated as:

$$\mathcal{R}^D = \frac{\mathcal{R}_{1,2} + \mathcal{R}_{3,2}}{2} + \mathcal{R}_{2,2}^D. \quad (24)$$

Also, the overall rate for our proposed M-frame using fixed target merging scheme is calculated as:

$$\mathcal{R}^M = \frac{\mathcal{R}_{1,2} + \mathcal{R}_{3,2}}{2} + \mathcal{R}_{2,2}^M. \quad (25)$$

The coding results are shown in Fig. 10 and BD-rate [33] comparison can be found in Table III. We observe from

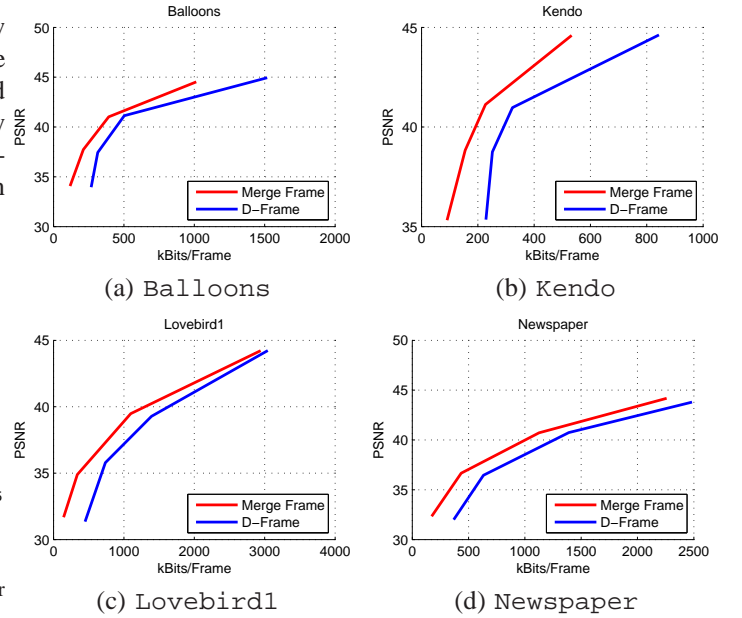


Fig. 10. PSNR v.s. encoding rate comparing proposed M-frame using fixed target merging scheme with D-frame for sequences Balloons, Kendo, Lovebird1 and Newspaper in static view switching scenario.

TABLE III
BD-RATE REDUCTION OF PROPOSED M-FRAME USING FIXED TARGET MERGING SCHEME COMPARED TO D-FRAME IN STATIC VIEW SWITCHING SCENARIO.

Sequence Name	M-frame vs. D-frame
Balloons	-31.7%
Kendo	-40.1%
Lovebird1	-35.7%
Newspaper	-31.1%

Table III that our proposed M-frame using fixed target merging scheme achieved up to 40.1% BD-rate reduction compared to D-frame. Further, from Fig. 10 we observe that our M-frame is better than D-frame in all bit-rate regions, especially in low and high bit-rate region, mainly due to the skip block and EOB flag tools. In high bit-rate region, due to the small distortion in SI frames, more blocks will be classified into skip block, which efficiently reduces the bits to encode the M-frame, while in low bit-rate region more coefficients are set to zero and skipped due to the EOB flag. This shows the effectiveness of our proposed M-frame using fixed target merging scheme compared to the D-frame.

D. Scenario 2: Bit-rate Adaptation

We next conducted experiments of bitrate adaptation scenario for single-view video sequences. M-frame is encoded in a RD-optimized manner, described in section VI with the system framework shown in Fig. 2. Three streams of different rates are encoded according to AIMD rate control behavior.

We constructed M- / D- frames to enable stream-switching from stream 1, 2 or 3 to target stream 2 under different bit-rates. We first encode three SI frames using $\Pi_{2,2}$ as target and $\Pi_{1,1}$, $\Pi_{2,1}$ and $\Pi_{3,1}$ as reference respectively. This results

in encoded rate $\mathcal{R}_{1,1}$, $\mathcal{R}_{2,1}$ and $\mathcal{R}_{3,1}$ for the three SI frames, respectively. Then we encoded a M- / D-frame to merge these three SI frames into an identical frame. The corresponding rate for M-frame and D-frame are $\mathcal{R}_{2,2}^M$ and $\mathcal{R}_{2,2}^D$, respectively.

We also constructed SP-frames to enable stream-switching from stream 1, 2 or 3 to target stream 2. We first encoded a primary SP-frame using $\Pi_{2,2}$ as target and $\Pi_{2,1}$ as reference. We then losslessly encoded two secondary SP-frames using the primary SP-frame as target and $\Pi_{1,1}$, $\Pi_{3,1}$ as reference respectively. $\mathcal{R}_{2,1}^S$ denotes the rate for primary SP-frame while $\mathcal{R}_{1,1}^S$ and $\mathcal{R}_{3,1}^S$ denote the rate for two secondary SP-frames.

As measure for transmission rate, we consider both the average and worst case code rate during a stream-switch. For average case, in the absence of application-dependent information, we assume that the probability of stream-switching is equal for all views. Thus, the overall rate for RD optimized M-frame is calculated as:

$$\mathcal{R}_{TA}^M = \frac{\mathcal{R}_{1,1} + \mathcal{R}_{2,1} + \mathcal{R}_{3,1}}{3} + \mathcal{R}_{2,2}^M. \quad (26)$$

The overall rate for D-frame is calculated as:

$$\mathcal{R}_{TA}^D = \frac{\mathcal{R}_{1,1} + \mathcal{R}_{2,1} + \mathcal{R}_{3,1}}{3} + \mathcal{R}_{2,2}^D. \quad (27)$$

The overall rate for SP-frame is calculated as:

$$\mathcal{R}_{TA}^{SP} = \frac{\mathcal{R}_{1,1}^S + \mathcal{R}_{2,1}^S + \mathcal{R}_{3,1}^S}{3}. \quad (28)$$

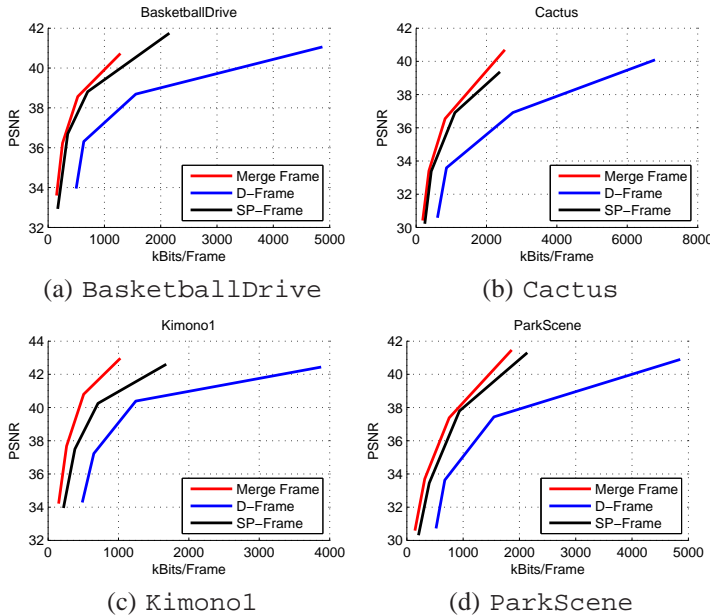


Fig. 11. PSNR versus encoding rate comparing proposed RD-optimized M-frame with D-frame and SP-frame for sequences BasketballDrive, Cactus, Kimono1 and ParkScene in average case.

The coding results of average case are shown in Fig. 11 and BD-rate comparison can be found in Table IV. We observe from Table IV that our proposed RD-optimized M-frame achieves up to 65.6% BD-rate reduction compared to D-frame and 36.3% BD-rate reduction compared to SP-frame. Moreover, from Fig. 11 we observe that our proposed RD-optimized M-frame is better than D-frame and SP-frame in

all bit-rate regions. Note that for the SP-frame case, if the switching probability to the primary SP-frame is higher, it will result in a smaller average rate.

For worst case, the code rate for M-frame is calculated as:

$$\mathcal{R}_{TW}^M = \max(\mathcal{R}_{1,1}, \mathcal{R}_{2,1}, \mathcal{R}_{3,1}) + \mathcal{R}_{2,2}^M. \quad (29)$$

The rate for D-frame is calculated as:

$$\mathcal{R}_{TW}^D = \max(\mathcal{R}_{1,1}, \mathcal{R}_{2,1}, \mathcal{R}_{3,1}) + \mathcal{R}_{2,2}^D. \quad (30)$$

The rate for SP-frame is calculated as:

$$\mathcal{R}_{TW}^{SP} = \max(\mathcal{R}_{1,1}^S, \mathcal{R}_{2,1}^S, \mathcal{R}_{3,1}^S). \quad (31)$$

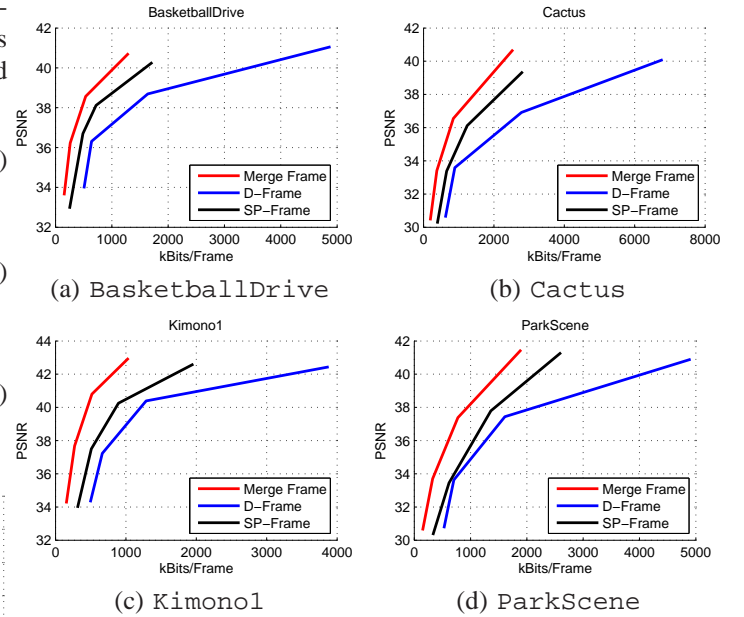


Fig. 12. PSNR versus encoding rate comparing RD-optimized M-frame with D-frame and SP-frame for sequences BasketballDrive, Cactus, Kimono1 and ParkScene in worst case.

The coding results of worst case are shown in Fig. 12 and BD-rate comparison can be found in Table IV. We observe from Table IV that our proposed RD-optimized M-frame achieves up to 65.4% BD-rate reduction compared to D-frame and 49.9% BD-rate reduction compared to SP-frame.

We observe in Table IV that the performance difference between average and worst case for D-frame is small. However, for SP-frame the performance difference between average and worst case is large. This is due to lossless coding in secondary SP-frames, resulting in a much larger size than primary SP-frame (typically 10 times larger).

E. Scenario 3: Dynamic View Switching

Finally we conducted experiments of dynamic view switching scenario for multiview video sequences. Three views are encoded using same QP . The detailed frame structure for M-frame, D-frame and SP-frame are the same as in Section VII-D. Also, the overall rate calculation for average and worst case are identical too.

The coding results of dynamic view switching for average case and worst case are shown in Fig. 13 and 14 respectively.

TABLE IV
BD-RATE REDUCTION OF RD-OPTIMIZED M-FRAME COMPARED TO D-FRAME AND SP-FRAME OF SCENARIO 2.

Sequence Name	M-frame vs. D-frame		M-frame vs. SP-frame	
	Average Case	Worst Case	Average Case	Worst Case
Balloons	-63.4%	-63.7%	-17.0%	-39.4%
Kendo	-63.5%	-63.2%	-18.8%	-42.1%
Lovebird1	-65.6%	-65.4%	-36.3%	-49.9%
Newspaper	-56.3%	-56.7%	-19.5%	-43.8%

TABLE V
BD-RATE REDUCTION OF RD-OPTIMIZED M-FRAME COMPARED TO D-FRAME AND SP-FRAME OF SCENARIO 3.

Sequence Name	M-frame vs. D-frame		M-frame vs. SP-frame	
	Average Case	Worst Case	Average Case	Worst Case
Balloons	-55.1%	-53.0%	-19.2%	-35.0%
Kendo	-53.8%	-53.6%	-19.3%	-36.4%
Lovebird1	-57.5%	-58.7%	-11.3%	-28.7%
Newspaper	-51.6%	-50.4%	-5.0%	-12.9%

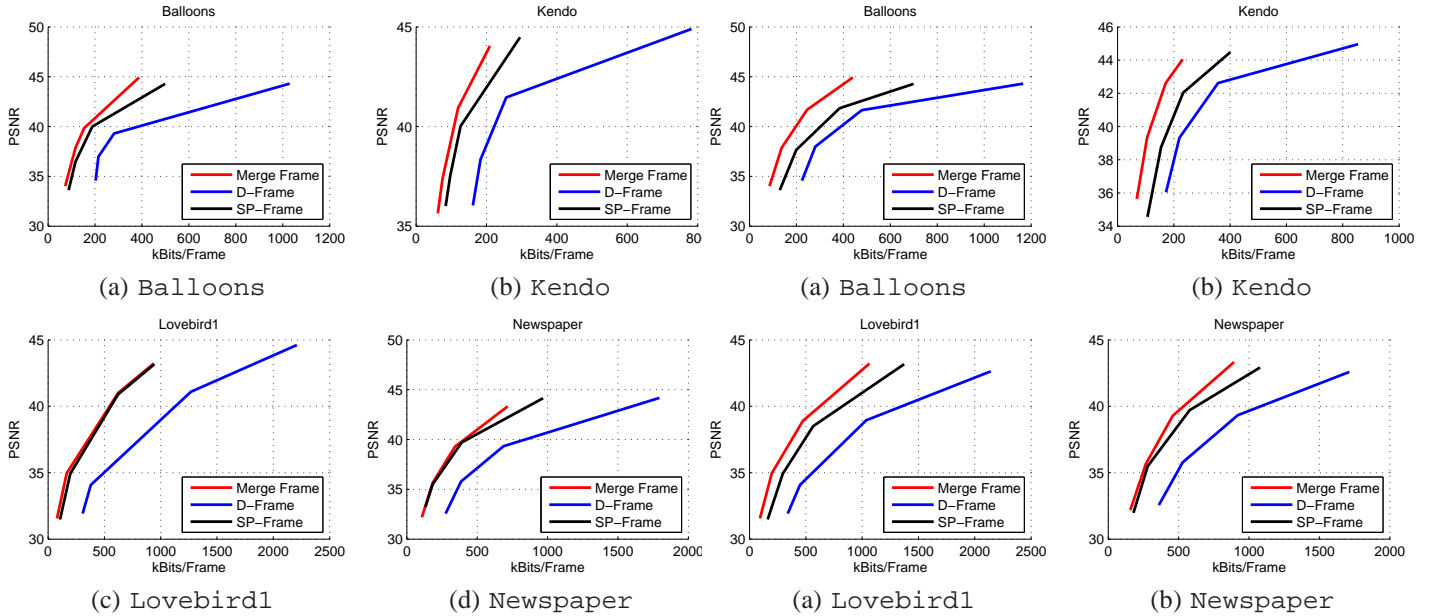


Fig. 13. PSNR versus encoding rate comparing proposed RD-optimized M-frame with D-frame and SP-frame for sequences Balloons, Kendo, Lovebird1 and Newspaper in average case.

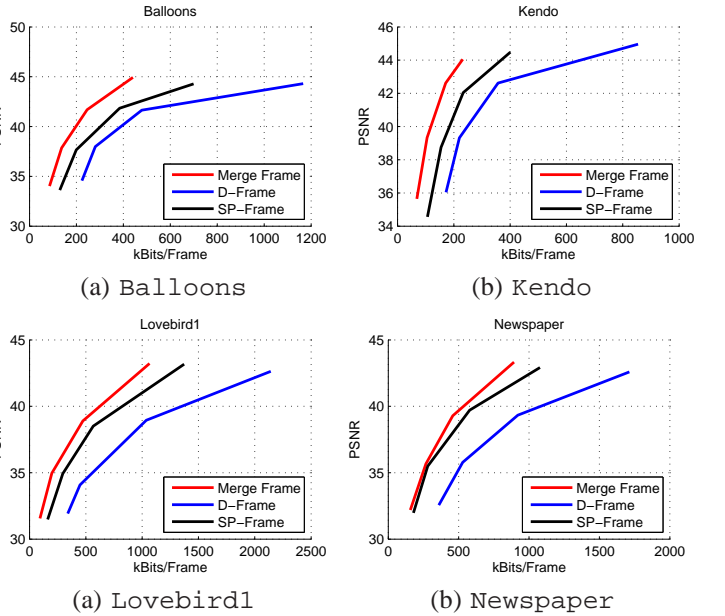


Fig. 14. PSNR versus encoding rate comparing proposed M-frame with D-frame and SP-frame for sequences Balloons, Kendo, Lovebird1 and Newspaper in worst case.

BD-rate comparison for average case and worst case can be found in Table V. From Table V we observe that our proposed RD-optimized M-frame achieves 57.5% BD-rate reduction compared to D-frame and 19.3% BD-rate reduction compared to SP-frame. From Table V we observe that our proposed RD-optimized M-frame achieves 58.7% BD-rate reduction compared to D-frame and 36.4% BD-rate reduction compared to SP-frame.

VIII. CONCLUSION

In this paper, we propose a new merging operator—piecewise constant (PWC) function—for merging different

reconstructed versions of a target frame to a unique one—to enable stream switching while preserving coding efficiency. Specifically, in order to merge k -th transform coefficients of different side information (SI) frames to the same value, we encode appropriate step sizes and horizontal shift parameters of a floor function, so that all the SI coefficients fall on the same function step. We propose two methods to select floor function parameters for signal merging. In the first method, we selected parameters so that coefficients are merged identically to a pre-determined target value. In the second method, the merged target value can be RD-optimized to induce better coding performance. Experimental results show that for both

cases, our proposed merge frame has significant coding gain over an implementation of DSC frame and H.264 SP-frames with a reduction in decoder complexity.

REFERENCES

- [1] G. Cheung, A. Ortega, N.-M. Cheung, and B. Girod, "On media data structures for interactive streaming in immersive applications," in *SPIE Visual Communications and Image Processing*, Huang Shan, China, July 2010.
- [2] M. Guo, Y. Lu, F. Wu, D. Zhao, and W. Gao, "Wyner-Ziv switching scheme for multiple bit-rate video streaming," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 5, May 2008, pp. 569–581.
- [3] G. Cheung, A. Ortega, and N.-M. Cheung, "Interactive streaming of stored multiview video using redundant frame structures," in *IEEE Transactions on Image Processing*, vol. 20, no. 3, March 2011, pp. 744–761.
- [4] —, "Generation of redundant coding structure for interactive multiview streaming," in *Seventeenth International Packet Video Workshop*, Seattle, WA, May 2009.
- [5] N.-M. Cheung, H. Wang, and A. Ortega, "Video compression with flexible playback order based on distributed source coding," in *IS&T/SPIE Visual Communications and Image Processing (VCIP'06)*, San Jose, CA, January 2006.
- [6] N.-M. Cheung and A. Ortega, "Compression algorithms for flexible video decoding," in *IS&T/SPIE Visual Communications and Image Processing (VCIP'08)*, San Jose, CA, January 2008.
- [7] N.-M. Cheung, A. Ortega, and G. Cheung, "Distributed source coding techniques for interactive multiview video streaming," in *27th Picture Coding Symposium*, Chicago, IL, May 2009.
- [8] W. Dai, G. Cheung, N.-M. Cheung, A. Ortega, and O. Au, "Rate-distortion optimized merge frame using piecewise constant functions," in *IEEE International Conference on Image Processing*, Melbourne, Australia, September 2013.
- [9] S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Transactions on Information Theory*, vol. 49, no. 3, pp. 626–643, 2003.
- [10] M. Karczewicz and R. Kurceren, "The SP- and SI-frames design for H.264/AVC," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, July 2003, pp. 637–644.
- [11] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, July 2003, pp. 560–576.
- [12] A. Aaron, P. Ramanathan, and B. Girod, "Wyner-Ziv coding of light fields for random access," in *IEEE International Workshop on Multimedia Signal Processing*, Siena, Italy, September 2004.
- [13] N.-M. Cheung and G. Cheung, "Coding for Interactive Navigation in High-dimensional Media Data," in *Emerging Technologies for 3D Video* (eds Frederic Dufaux, Beatrice Pesquet-Popescu, Marco Cagnazzo), 2013.
- [14] D. Taubman and R. Rosenbaum, "Rate-distortion optimized interactive browsing of JPEG2000 images," in *Proceedings of International Conference on Image Processing*, 2003, 2003, pp. 765–768.
- [15] I. Bauermann and E. Steinbach, "RDTC optimized compression of image-based scene representation (part I): modeling and theoretical analysis," *IEEE Transactions on Image Processing*, vol. 17, no. 5, pp. 709–723, 2008.
- [16] —, "RDTC optimized compression of image-based scene representation (part II): practical coding," *IEEE Transactions on Image Processing*, vol. 17, no. 5, pp. 724–736, 2008.
- [17] Z. Fan and A. Ortega, "Overlapped tiling for fast random access of 3-d datasets," in *Proceedings of the IEEE DCC'07*, 2009.
- [18] S. J. Wee and B. Vasudev, "Compressed-domain reverse play of MPEG video streams," in *Proceedings of SPIE Multimedia Systems and Applications* (eds A. G. Tescher, B. Vasudev, V. M. Bove, Jr., and B. Derryberry), 1999, pp. 237–248.
- [19] C. W. Lin, J. Zhou, J. Youn, and M. T. Sun, "MPEG video streaming with VCR functionality," *IEEE Trans. Circ. Syst. Vid.*, vol. 11, no. 3, pp. 415–425, 2001.
- [20] C.-H. Fu, Y.-L. Chan, and W.-C. Siu, "Efficient reverse-play algorithms for MPEG video with VCR support," *IEEE Trans. Circ. Syst. Vid.*, vol. 16, no. 1, pp. 19–30, 2006.
- [21] F.-O. Devaux, J. Meessen, C. Parisot, J. Delaigle, B. Macq, and C. D. Vleeschouwer, "A flexible video transmission system based on jpeg2000 conditional replenishment with multiple references," in *Proceedings of the IEEE ICASSP'07*, 2007.
- [22] A. Naman and D. Taubman, "JPEG2000-based scalable interactive video (JSIV)," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1435–1449, 2011.
- [23] A. Mavlankar and B. Girod, "Spatial-random-access-enabled video coding for interactive virtual pan/tilt/zoom functionality," *IEEE Trans. Circ. Syst. Vid.*, vol. 21, no. 5, pp. 577–588, 2011.
- [24] A. Mavlankar, P. Agrawal, D. Pang, S. Halawa, N.-M. Cheung, and B. Girod, "An interactive region-of-interest video streaming system for online lecture viewing," in *Proceedings of 18th International Packet Video Workshop (PV)*, 2010, pp. 64–71.
- [25] S. Halawa, D. Pang, N.-M. Cheung, and B. Girod, "ClassX: an open source interactive lecture streaming system," in *MM11 Proceedings of the ACM International Conference on Multimedia*, 2011, pp. 719–722.
- [26] D. Pang, S. Halawa, N.-M. Cheung, and B. Girod, "Mobile interactive region-of-interest video streaming with crowd-driven prefetching," in *Proceedings of IMPD 11 Proceedings of the 2011 International ACM Workshop on Interactive Multimedia on Mobile and Portable Devices*, 2011, pp. 7–12.
- [27] H. Huang, G. Chan, G. Cheung, and P. Frossard, "Distributed content replication for multiple movies in interactive multiview video streaming," in *19th International Packet Video Workshop*, Munich, Germany, May 2012.
- [28] J.-G. Lou, H. Cai, and J. Li, "A real-time interactive multi-view video system," in *ACM International Conference on Multimedia*, Singapore, November 2005.
- [29] G. J. Sullivan, "Efficient scalar quantization of exponential and laplacian random variables," *Information Theory, IEEE Transactions on*, vol. 42, no. 5, pp. 1365–1374, 1996.
- [30] N.-M. Cheung, A. Ortega, and G. Cheung, "Rate-distortion based reconstruction optimization in distributed source coding for interactive multiview video streaming," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*. IEEE, 2010, pp. 3721–3724.
- [31] D.-M. Chiu and R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks," *Computer Networks and ISDN systems*, vol. 17, no. 1, pp. 1–14, 1989.
- [32] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.
- [33] G. Bjontegaard, "Improvements of the BD-PSNR model," *document VCEG-A111, ITU-T SG16*, 2008.